

修 士 論 文 概 要 書

2010 年 2 月提出

専攻名	情報・ネットワーク専攻	氏名	巻島 一雄	指 導	中島 達夫	印
研究指導名	分散システム研究	学籍番号	5108B116 - 1 ^{CD}	教 員		
研 究 題 目	リアルタイム OS と汎用 OS の共存する SMP 環境における仮想 CPU マイグレーション					

1 序論

1.1 背景

近年、我々が日常生活で用いる組み込み機器は、非常に高機能化している。携帯電話は本来の電話機能に加えてウェブブラウザ、ワンセグの受信・再生機能などを、音楽プレーヤーはゲームや動画再生の機能を備えるようになった。これらの組み込み機器はデスクトップ PC に近い高機能を求められながらも、バッテリーの駆動時間を長くするために電力の消費を抑える事が求められる。そして、SMP を採用する事でシングルプロセッサよりも消費電力を抑えられるケースがある事が報告されている [2]。したがって消費電力を抑えつつ高機能を実装するという要求に応えるため、組み込み機器でも SMP を採用する例が今後増えていくと予想される。

エンジニアリングコストを低減するために、マルチ OS 環境を組み込み機器の開発に採用する例が存在する。例えばマルチメディア処理やネットワーク処理を実現するために、既に膨大なオープンソースのパッケージが存在する Linux のような汎用 OS を開発に用いる。そしてリアルタイム性を要求する処理を、従来開発に用いてきたリアルタイム OS (以下、RTOS) のソフトウェア資産を引き続き利用して実装するという開発手法である。しかし異なる OS を 1 つのマシン上で動かすためには、通常はそれぞれの OS に最低 CPU を 1 つ占有させなければならず、製品のコスト増を招くという欠点がある。

我々は過去に、SMP 上でリアルタイム OS と汎用 OS を同時に動作させるための CPU 仮想化層、SPUMONE を開発した [1]。これはリアルタイム OS と汎用 OS の間で CPU を共有できるようにするソフトウェアである。汎用 OS は余剰の CPU パワーを利用できるため、共有しないケースと比較してパフォーマンスの向上が達成できた。しかし CPU 仮想化層は汎用 OS のカーネルを特権レベルで実行する事を許しており、これがリアルタイム OS の動作を阻害するという問題を引き起こした。すなわち汎用 OS がリアルタイム OS への割り込みを禁止して、リアルタイム性を損ねるという問題である。

1.2 提案

本研究で提案する仮想 CPU マイグレーションの機能を導入すると、CPU 仮想化層は汎用 OS の仮想 CPU を、ある物理 CPU から別の物理 CPU に移動して動作させる事ができるようになる。これを用いて CPU 仮想化層は、汎用 OS の仮想 CPU が特権レベルに移行する時に他の物理 CPU に移動させ、ユーザーレベルに移行する時に共有 CPU 上に戻す操作を行う。以上の操作を行う事で、汎用 OS による割り込み禁止の区間は最小限に抑えられる。

2 設計と実装

2.1 システム構成

システム構成の概要を図 1 に示す。仮想 CPU マイグレーションによって他の物理 CPU に移動する仮想 CPU は、静的に決まっており、これを簡単のためイミгранトと呼称する。また移動先となる物理 CPU も静的に決まっている。

2.2 マイグレーション・ポリシー

- 共有 CPU 上のイミгранトに配送すべき例外・割り込み・TLB ミス例外を SPUMONE が受け取ったら、SPUMONE はイミгранトを物理 CPU2 にマイグレーションする。
- 物理 CPU2 上にイミгранトが存在する時、イミгранトがカーネルからユーザー空間に移行する時は、SPUMONE はイミгранトを共有 CPU にマイグレーションする。

2.3 物理 CPU2 へのマイグレーション

共有 CPU 上の SPUMONE は、マイグレーション操作の始めにプロセッサ間割り込みを物理 CPU2 に送信する。その後レジスタの保存作業、キャッシュのフラッシュを開始する事で、移行にかかる時間を最小限に抑えている。

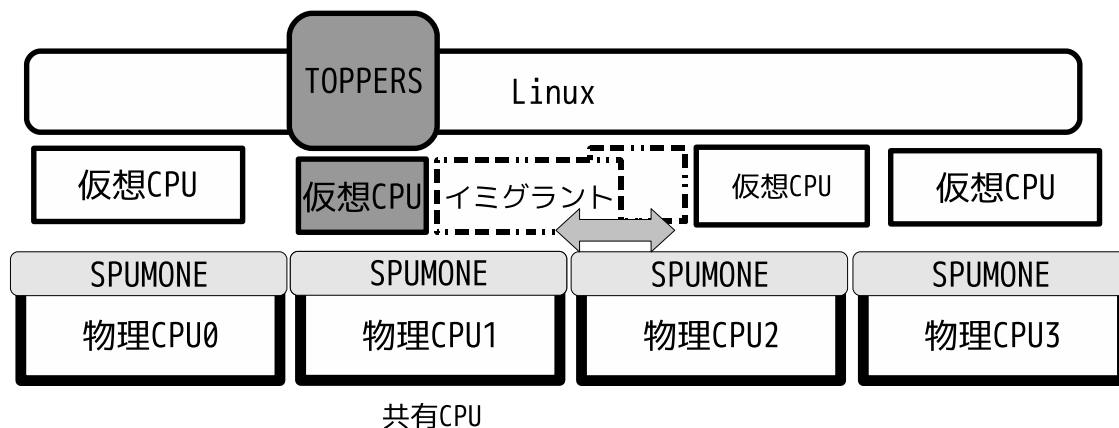


図 1: システム構成

2.4 共有 CPU に帰還するマイグレーション

物理 CPU2 上の SPUMONE は、マイグレーション操作の最後にプロセッサ間割込みを共有 CPU に送信する。その前にレジスタの保存作業、キャッシュのフラッシュを完了しておく事で、共有 CPU 上の RTOS が停止する時間を最小限に抑えている。

3 計測結果

表 1 に RTOS の割込み遅延の計測結果を、表 2 に Dhrystone による Linux の性能の計測結果を示す。

Linux における操作	遅延 (μ s)
ディスク + ネットワークアクセスによる負荷	9.68
プロセッサ間割込みによる負荷	14.98
イミグランドによる 10μ s の割込み禁止	6.96
イミグランドによる 100μ s の割込み禁止	7.72
イミグランドによる 1000μ s の割込み禁止	7.18

表 1: RTOS の割込み遅延

システム構成	RTOS の負荷	DMIPS 値
Linux 3CPU	なし	1498
SPUMONE + Linux 4CPU	なし	1954
	20%	1789
	40%	1598
	60%	1384
	80%	1207

表 2: Linux の性能

4 将来課題

RTCPU 上ではユーザーモードでしか Linux は動作しないため、実行時間の内の大部分をカーネル内で過ごすよ

うなスレッドは RTCPU に割り当てられるべきではない。例えばファイルアクセスが主体のスレッドなどである。したがって、どの CPU が RTCPU であるかを SPUMONE が汎用 OS に通知して、汎用 OS 自身でスレッドのアカウントティングを行い、ユーザーモードで過ごす時間が長いスレッドを RTCPU に積極的に割り当てるようにするというスケジューリングを行うように、汎用 OS のスケジューラを改変しなければならない。これは今後の課題と考えられる。

5 結論

本研究では、我々は仮想 CPU マイグレーションの機能を SH4A の 4 コアを装備した SMP 上に実装し、評価した。その結果、リアルタイム OS の割込み遅延は最大で約 15μ s に抑えられる事を保証できた。仮想 CPU マイグレーションによるリアルタイム OS の保護は、汎用 OS のカーネルによって影響を受ける事がない。したがって、どのようなデバイスドライバやカーネル・モジュールを汎用 OS に実装しても、リアルタイム OS のリアルタイム性に悪影響を及ぼすことはない。この性質から、汎用 OS の開発の結果、RTOS に悪影響が現れ、そこで開発が滞るという問題は回避されるだろうと期待できる。また我々は、リアルタイム OS の CPU 使用率が高くない状態では、汎用 OS がより多くの CPU 資源を活用できる事を確認した。

参考文献

- [1] Wataru Kanda. Design and implementation of a multi-core cpu virtualization technique for embedded systems. Master's thesis, Waseda University, 2009.
- [2] Idehara Akio, Tawara Yasuhiro, Yamamoto Hitoshi, Sugai Naoto, and Iizuka Tsuyoshi. An evaluation of dynamic power management support of smp linux for embedded multicore processor. In *Embedded Systems Symposium 2008*, pp. 115–123.